

JACOBSON, PRICE, HOLMAN & STERN

PROFESSIONAL LIMITED LIABILITY COMPANY

THE JENIFER BUILDING

400 SEVENTH STREET, N. W.

WASHINGTON, D. C. 20004

(202) 638-6666

YOON S. HAM

Direct: (202) 662-8483

yham@jphs.com

April 17, 2001

Atty. Docket No.: **11349-P66590US0****CUSTOMER NUMBER: 00136**

HARVEY B. JACOBSON, JR.
D. DOUGLAS PRICE
JOHN CLARKE HOLMAN
SIMOR L. MOSKOWITZ
ALLEN S. MELSER
MICHAEL R. SLOBASKY
MARSHA G. GENTNER
JONATHAN L. SCHERER
IRWIN M. AISENBERG
GEORGE W. LEWIS
WILLIAM E. PLAYER
YOON S. HAM
PHILIP L. O'NEILL
LINDA J. SHAPIRO
LEESA N. WEISS
SUZIN C. BAILEY
MATTHEW J. CUCCIAS
DANIEL K. DORSEY
LEIGH M. ZANOWSKI
MONICA LEWIS

OF COUNSEL
MARVIN R. STERN
NATHANIEL A. HUMPHRIES

TELEFAX:
(202) 393-5350
(202) 393-5351
(202) 393-5352

E-MAIL: IP@JPHS.COM
INTERNET: WWW.JPHS.COM

*BAR OTHER THAN DC

10997 U.S. Pat. 09/835528



Honorable Commissioner for Patents

Washington, D.C. 20231

Sir:

Transmitted herewith for filing is the patent application in the names of:

Woo-Jin LEE of Taejon, Republic of Korea;
Min-Jeong KIM of Yongin-Shi, Republic of Korea; and
Gyu-Sang SHIN of Taejon, Republic of Korea,

for **METHOD AND APPARATUS FOR IDENTIFYING SOFTWARE COMPONENTS USING OBJECT RELATIONSHIPS AND OBJECT USAGES IN USE CASES** The application comprises a 18-page specification including 9 claims (3 independent) and Abstract, 4 sheets of drawings (Figs. 1-5), and a Declaration and Power of Attorney.

The above-mentioned applicants qualify for small entity status.

Accompanying this application for filing are:

- (1) Assignment document, cover sheet and **\$40.00** fee for recordation of Assignment;
- (2) Information Disclosure Statement, PTO-1449 form and copies of references; and
- (3) A certified copy of **Korean** Application No. **2000-76642**, filed **December 14, 2000**, the priority of which is claimed under 35 U.S.C. §119.

The filing fee has been calculated as shown:

| | | |
|--------------------------------|-----------------------------------|------------------|
| Small Entity | | \$ 355.00 |
| Total Claims= <u>09</u> ; | in excess of 20 = 0 x (\$09.00) = | |
| Total Ind. Claims= <u>03</u> ; | in excess of 03 = 0 x (\$40.00) = | + |
| TOTAL FILING FEE: | | \$ 355.00 |

A check in the amount of **\$395.00** is enclosed to cover the total Filing Fee and an Assignment Recordation Fee. The Commissioner is hereby authorized to charge payment of any fees set forth in Sections 1.16 or 1.17 during the pendency of this application, or credit any overpayment, to deposit Account No. 06-1358. A duplicate copy of this sheet is enclosed.

Respectfully submitted,

By: 

Yoon S. Ham, Reg. No. 45,307

Enclosures

YSH:ecl

<Priority Document Translation>



THE KOREAN INDUSTRIAL
PROPERTY OFFICE

This is to certify that the following application annexed
hereto is a true copy from the records of the Korean Industrial
Property Office.

Application Number : 2000-76642 (Patent)

Date of Application : December 14, 2000

Applicant(s) : ELECTRONICS AND TELECOMMUNICATIONS
RESEARCH INSTITUTE

April 4, 2001

COMMISSIONER

JC997 U.S. PTO
09/835528
04/17/01

대한민국 특허청

KOREAN INTELLECTUAL PROPERTY OFFICE

별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto
is a true copy from the records of the Korean Intellectual
Property Office.

출원번호 :
Application Number

특허출원 2000년 제 76642 호

출원년월일 :
Date of Application

2000년 12월 14일

출원인 :
Applicant(s)

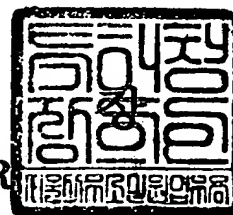
한국전자통신연구원



2001 년 04 월 04 일

특 허 청

COMMISSIONER



| | |
|------------|---|
| 【서류명】 | 특허출원서 |
| 【권리구분】 | 특허 |
| 【수신처】 | 특허청장 |
| 【참조번호】 | 0003 |
| 【제출일자】 | 2000.12.14 |
| 【발명의 명칭】 | 객체 관계와 객체 이용 정보를 이용한 소프트웨어 컴포넌트 식별 장치 및 그 방법 |
| 【발명의 영문명칭】 | Method and apparatus for identifying software components using object relationships and object usage in use cases |
| 【출원인】 | |
| 【명칭】 | 한국전자통신연구원 |
| 【출원인코드】 | 3-1998-007763-8 |
| 【대리인】 | |
| 【성명】 | 특허법인 신성 정지원 |
| 【대리인코드】 | 9-2000-000292-3 |
| 【포괄위임등록번호】 | 2000-051975-8 |
| 【대리인】 | |
| 【성명】 | 특허법인 신성 원석희 |
| 【대리인코드】 | 9-1998-000444-1 |
| 【포괄위임등록번호】 | 2000-051975-8 |
| 【대리인】 | |
| 【성명】 | 특허법인 신성 박해천 |
| 【대리인코드】 | 9-1998-000223-4 |
| 【포괄위임등록번호】 | 2000-051975-8 |
| 【발명자】 | |
| 【성명의 국문표기】 | 이우진 |
| 【성명의 영문표기】 | LEE, Woo Jin |
| 【주민등록번호】 | 691107-1772314 |
| 【우편번호】 | 302-150 |
| 【주소】 | 대전광역시 서구 만년동 281 상록수아파트 103-803 |
| 【국적】 | KR |

【발명자】**【성명의 국문표기】**

김민정

【성명의 영문표기】

KIM,Min Jeong

【주민등록번호】

751012-2167912

【우편번호】

449-840

【주소】

경기도 용인시 수지읍 풍덕천694 건영아파트 101-1103

【국적】

KR

【발명자】**【성명의 국문표기】**

신규상

【성명의 영문표기】

SHIN,Gyu Sang

【주민등록번호】

541227-1018134

【우편번호】

300-200

【주소】

대전광역시 동구 용전동 1 신동아아파트 11-903

【국적】

KR

【심사청구】

청구

【취지】

특허법 제42조의 규정에 의한 출원, 특허법 제60조의 규정에 의한 출원심사를 청구합니다. 대리인
 특허법인 신성 정지원 (인) 대리인
 특허법인 신성 원석희 (인) 대리인
 특허법인 신성 박해천 (인)

【수수료】**【기본출원료】**

20 면 29,000 원

【가산출원료】

1 면 1,000 원

【우선권주장료】

0 건 0 원

【심사청구료】

9 항 397,000 원

【합계】

427,000 원

【감면사유】

정부출연연구기관

【감면후 수수료】

213,500 원

【첨부서류】

1. 요약서·명세서(도면)_1통

【요약서】**【요약】**

본 발명은 객체 관계와 객체 이용 정보를 이용한 소프트웨어 컴포넌트 식별 장치 및 그 방법에 관한 것으로, 객체 관계와 객체 이용 정보를 이용하여 소프트웨어 컴포넌트를 식별하기 위한 소프트웨어 컴포넌트 식별 장치 및 그 방법과 상기 방법을 실현시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체를 제공하기 위하여, 객체 관계와 객체 이용 정보를 이용한 소프트웨어 컴포넌트 식별 장치에 있어서, 사용자로부터 정보를 입력받기 위한 사용자 인터페이싱 수단; 상기 사용자 인터페이싱 수단으로부터 정보를 입력받아 객체간의 의존 가중치를 계산하기 위한 가중치 계산수단; 상기 사용자 인터페이싱 수단으로부터 입력받은 정보와 상기 가중치 계산수단에서 구해진 객체간의 의존 가중치를 이용하여 객체 의존 네트워크를 생성하기 위한 객체 네트워크 생성수단; 및 상기 객체 네트워크 생성수단에서 생성된 객체 의존 네트워크를 이용하여 컴포넌트 식별 과정을 제어하기 위한 컴포넌트 식별수단을 포함하며, 컴포넌트 기반 소프트웨어 개발 도구 등에 이용됨.

【대표도】

도 4

【색인어】

객체 관계, 객체 이용 정보, 컴포넌트 식별, 의존 지수, 중요 지수, 가중치 값

【명세서】**【발명의 명칭】**

객체 관계와 객체 이용 정보를 이용한 소프트웨어 컴포넌트 식별 장치 및 그 방법
{Method and apparatus for identifying software components using object relationships
and object usages in use cases}

【도면의 간단한 설명】

도 1 은 일반적인 컴포넌트 기반 소프트웨어 개발 과정에 대한 일실시에 흐름도.

도 2 는 본 발명에 따른 객체 및 이용 케이스(Use Case) 연관 그래프에 대한 일실시에 설명도.

도 3 은 본 발명에 따른 객체 의존성 네트워크에 대한 일실시에 설명도.

도 4 는 본 발명에 따른 컴포넌트 식별 장치의 일실시에 구성도.

도 5 는 본 발명에 따른 소프트웨어 컴포넌트 식별 방법에 대한 일실시에 흐름도.

* 도면의 주요 부분에 대한 부호의 설명

410 : 사용자 인터페이스 420 : 가중치 계산기

430 : 객체 네트워크 생성기 440 : 컴포넌트 식별기

【발명의 상세한 설명】**【발명의 목적】****【발명이 속하는 기술분야 및 그 분야의 종래기술】**

- <9> 본 발명은 객체 관계와 객체 이용 정보를 이용한 소프트웨어 컴포넌트 식별 장치 및 그 방법과 상기 방법을 실현시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체에 관한 것으로, 더욱 상세하게는 객체 지향 영역 모델링 과정에서 얻어지는 객체 모델과 각 이용 케이스(Use Case)별 순차도 정보에서 추출한 객체간의 관계 정보 및 누적 객체 이용 정보를 이용하여 객체 의존성 네트워크를 생성하고, 상기 네트워크에서의 객체 관계 유형과 가중치를 이용하여 객체들을 컴포넌트로 묶어서 식별하는 소프트웨어 컴포넌트 식별 장치 및 그 방법과 상기 방법을 실현시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체에 관한 것이다.
- <10> 컴포넌트 기반 개발 방법론은 객체 지향 방법론에서 거론된 몇 가지 문제점들을 해결하기 위해 도입된 개념으로 기본적인 객체 지향 개념은 포함하고 있다고 보아도 무방하다.
- <11> 도 1 은 일반적인 컴포넌트 기반 소프트웨어 개발 과정에 대한 일실시에 흐름도이다.
- <12> 먼저, 대상 시스템을 명확히 이해하기 위해 유엠엘(UML : Unified Modeling Language) 모델링 기법을 이용한 영역 모델링을 수행한다(101). 그리고, 이 정보를 바탕으로 재사용성이 높고, 독립적으로 수행 가능한 컴포넌트 단위들을 식별한다(102).
- <13> 다음으로, 컴포넌트 다이어그램을 이용하여 각 컴포넌트가 제공하는 인터페이스와

다른 컴포넌트와의 의존성 등을 명확히 기술한다(103). 이는, 컴포넌트 다이어그램의 인터페이스를 유지하면 각 컴포넌트별로 독립적인 개발이 가능하기 때문이다. 그러므로, 컴포넌트별로 그리고 다른 컴포넌트와 병행적으로 내부 상세 설계 및 구현 등의 과정을 거쳐 이진(binary) 컴포넌트 코드를 생성하게 된다(104).

<14> 한편, 컴포넌트 기반 소프트웨어 개발 방법의 여러 단계에서 핵심적인 부분 중 하나로 컴포넌트 식별 부분을 들 수 있다.

<15> 상기 컴포넌트 식별 과정은 시스템이 속하는 영역(domain)의 특성과 영역 정보들을 바탕으로 컴포넌트의 특성인 재사용성, 독립성 등을 만족하는 부분을 선별한다.

<16> 그러나, 영역(domain) 정보에는 비계량적인 요소가 많고 재사용성, 독립성 등의 기준 자체 또한 명확하지 않아 주로 영역 전문가의 경험에 의한 직관에 의존하여 왔다.

<17> 지금까지는 영역 전문가(domain expert)들이 사용해 온 컴포넌트 식별 방법으로는 객체 모델에 나타난 객체 관계 정보를 바탕으로 연관성이 높은 객체들을 묶어 컴포넌트화하는 방식, 프로세스 설계 관점에서 순차도(sequence diagram)에 나타난 제어 흐름 등을 고려하여 하나의 작업 스레드(thread) 별로 컴포넌트를 식별하는 방식, 소프트웨어 아키텍처 레벨에서 서브 시스템으로 식별된 부분을 컴포넌트로 묶는 방법 등이 있다.

<18> 상기 방법들은 영역 모델링 과정에서 생성되는 객체 관점, 프로세스 관점, 아키텍처 관점 등의 여러 관점의 정보 중 특정 관점 하나만을 고려하여 컴포넌트를 식별하고 있으며, 또한 제시된 컴포넌트 식별 기준이 모호하기 때문에 적용하는 사람에 따라 다양한 결과가 나오게 되므로, 정확히 컴포넌트를 식별할 수 없는 문제점이 있었다.

【발명이 이루고자 하는 기술적 과제】

<19> 본 발명은, 상기한 바와 같은 문제점을 해결하기 위하여 제안된 것으로, 객체 관계와 객체 이용 정보를 이용하여 소프트웨어 컴포넌트를 식별하기 위한 소프트웨어 컴포넌트 식별 장치 및 그 방법과 상기 방법을 실현시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체를 제공하는데 그 목적이 있다.

<20> 즉, 본 발명은, 객체 지향 영역 모델링 과정에서 얻어지는 객체 모델과 각 이용 케이스(Use Case)별 순차도 정보에서 추출한 객체간의 관계 정보 및 누적 객체 이용 정보를 이용하여 객체 의존성 네트워크를 생성하고, 상기 네트워크에서의 객체 관계 유형과 가중치를 이용하여 객체들을 컴포넌트로 묶어서 식별하는 소프트웨어 컴포넌트 식별 장치 및 그 방법과 상기 방법을 실현시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체를 제공하는데 그 목적이 있다.

【발명의 구성 및 작용】

<21> 상기 목적을 달성하기 위한 본 발명의 장치는, 객체 관계와 객체 이용 정보를 이용한 소프트웨어 컴포넌트 식별 장치에 있어서, 사용자로부터 정보를 입력받기 위한 사용자 인터페이싱 수단; 상기 사용자 인터페이싱 수단으로부터 정보를 입력받아 객체간의 의존 가중치를 계산하기 위한 가중치 계산수단; 상기 사용자 인터페이싱 수단으로부터 입력받은 정보와 상기 가중치 계산수단에서 구해진 객체간의 의존 가중치를 이용하여 객체 의존도 네트워크를 생성하기 위한 객체 네트워크 생성수단; 및 상기 객체 네트워크 생성수단에서 생성된 객체 의존도 네트워크를 이용하여 컴포넌트 식별 과정을 제어하기

위한 컴포넌트 식별수단을 포함하는 것을 특징으로 한다.

<22> 한편, 본 발명의 방법은, 소프트웨어 컴포넌트 식별 장치에 적용되는 컴포넌트 식별 방법에 있어서, 객체 모델 정보를 입력받아 유스 케이스(이용 케이스) 및 객체 의존성 그래프를 생성하는 제 1 단계; 상기 유스 케이스(이용 케이스)에서 추출된 객체 이용 정보에 대하여 객체 의존 지수를 계산하고, 객체에 연결된 입력 아크(incoming arc)의 가중치를 더하여 객체 중요 지수를 계산하는 제 2 단계; 상기 객체 중요 지수 값이 사용자가 미리 정의한 특정 임계 값(Importance Degree Threshold)을 넘는 객체를 씨드(Seed) 객체로 결정하여 컴포넌트에 설정하는 제 3 단계; 및 컴포넌트와의 의존성 값이 특정 임계값 이상이 되는 객체를 컴포넌트에 추가하는 과정을, 추가로 네비게이션이 가능한 객체에 대하여 수행하는 제 4 단계를 포함하는 것을 특징으로 한다.

<23> 한편, 본 발명은, 대용량 프로세서를 구비한 소프트웨어 컴포넌트 식별 장치에, 객체 모델 정보를 입력받아 유스 케이스(이용 케이스) 및 객체 의존성 그래프를 생성하는 제 1 기능; 상기 유스 케이스(이용 케이스)에서 추출된 객체 이용 정보에 대하여 객체 의존 지수를 계산하고, 객체에 연결된 입력 아크(incoming arc)의 가중치를 더하여 객체 중요 지수를 계산하는 제 2 기능; 상기 객체 중요 지수 값이 사용자가 미리 정의한 특정 임계 값(Importance Degree Threshold)을 넘는 객체를 씨드(Seed) 객체로 결정하여 컴포넌트에 설정하는 제 3 기능; 및 컴포넌트와의 의존성 값이 특정 임계값 이상이 되는 객체를 컴포넌트에 추가하는 과정을, 추가로 네비게이션이 가능한 객체에 대하여 수행하는 제 4 기능을 실현시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체를 제공한다.

<24> 이처럼 본 발명에서는, 영역 모델링 정보인 객체 모델과 이용 케이스별 순차도

(sequence diagram)에 나타난 정보를 바탕으로 객체들 간의 구조적인 연관관계 및 객체 이용 관계를 기술하고, 이를 이용하여 양의 실수 값으로 표현되는 객체 의존 지수(Dependency Degree)와 객체 중요 지수(Importance Degree)를 나타내는 객체 의존도 네트워크(object dependency network)을 생성한다. 그리고, 상기 객체 의존도 네트워크에 나타난 객체간의 의존 정도, 객체의 중요 지수, 의존 유형 등을 바탕으로 컴포넌트의 독립성, 재사용성 등을 고려하여 컴포넌트로 나뉘어질 수 있는 객체들로 클러스터링(clustering)을 수행하는 것을 특징으로 한다.

- <25> 상술한 목적, 특징들 및 장점은 첨부된 도면과 관련한 다음의 상세한 설명을 통하여 보다 분명해 질 것이다. 이하, 첨부된 도면을 참조하여 본 발명에 따른 바람직한 일 실시예를 상세히 설명한다.
- <26> 도 2 는 본 발명에 따른 객체 및 이용 케이스 연관 그래프에 대한 일 실시예 설명도이다.
- <27> 도 2 에 도시된 바와 같이, 객체 및 이용 케이스 연관 그래프는 이용 케이스(210, 211)와 객체(220~224)를 각각 노드(node)로 나타내고, 상기 객체 및 이용 케이스간의 의존 관계는 구조적 관점의 키워드인 일반화(Generalization), 분해(Decomposition)와 객체 이용 관점의 키워드인 생성/파괴(Create/Destroy : CD), 생성(Create : C), 갱신(Update : U), 참조(Reference : R)의 6개의 의존 관계 키워드로 기술된다.
- <28> 상기 이용 케이스(210, 211)와 객체(220, 224)는 서로 다른 모양으로 기술되며, 노드간의 의존성의 유형에는 일반화(Generalization, 230)와 같은 구조적 관계 정보와 생성/파괴(Create/Destroy)(231~233)의 객체 사용 정보로 나눌 수 있다.
- <29> 상기 객체 사용 정보(231~233)는 이용 케이스(210, 211)별로 나타나는 것이므로, 여러

이용 케이스(210, 211)에 연관된 객체의 경우는 갱신 + 참조(Update + Reference)(235)와 같이 누적된 객체 이용 정보를 기술한다. 그리고, 각 이용 케이스 노드 옆에는 실수 값(240~241)이 부가되어 있는데, 이는 이용 케이스(210, 211)의 중요도를 0.1 ~ 1.0 사이 값으로 표현한 가중치 값이다.

<30> 도 3 은 본 발명에 따른 객체 의존성 네트워크에 대한 일실예 설명도로서, 객체와 이용 케이스간의 의존성을 제하고, 객체간의 의존성을 양의 실수로 표현한 의존 지수(DD)와 이를 이용하여 계산된 각 객체의 중요 지수(ID)를 나타내고 있는 객체 의존성 네트워크를 보여준다.

<31> 도 3 에 도시된 바와 같이, 객체(311~315)는 다른 객체와의 의존 지수를 1.0(321)과 같이 양의 실수 값으로 나타낸다. 이러한 의존 지수(321~324)의 계산은 이용 케이스 및 객체 의존 그래프에 나타난 이용 케이스의 가중치, 객체간의 구조적 관계정보, 누적 객체 이용 정보들로부터 계산된다.

<32> 아래 [표 1]은 의존 유형(Dependency Type)에 따라 의존 값으로 변환하는 매핑(mapping) 테이블의 일 예를 보인다.

<33> 【표 1】

| 의존 유형 Dependency Type(DT) | 가중치 값 Weight(DT) |
|------------------------------|---------------------|
| Aggregation | 1.0 |
| Decomposition | 1.0 |
| Create/Destroy | 0.9 |
| Create | 0.7 |
| Update | 0.5 |
| Reference | 0.3 |

<34> 또한, 아래 [수학식 1]은 객체간의 각 의존성 가중치를 계산하는 식이다. 의존성 가중치는 이용 케이스별로 사용된 각 객체의 의존 유형에 이용 케이스의 가중치를 곱하여 더한 것이다.

<35> 【수학식 1】

$$\sum_{i=1}^n W(UC\ i) * Weight(DT)$$

<36> 상기 객체(311~315)를 나타내는 원 안에 표시된 가중치 값(331~335)은 객체의 중요지수(ID)를 나타낸다.

<37> 상기 중요 지수(ID) 값은 객체로 들어오는 아크(incoming arc)에 있는 의존 지수들을 더한 값이다. 즉, 특정 객체가 다른 객체들에게 연관되어 있는 가중치를 모두 더하여 그 객체의 중요도를 평가한 것이다.

<38> 도 4 는 본 발명에 따른 컴포넌트 식별 장치의 일실시에 구성도이다.

<39> 도 4 에 도시된 바와 같이, 본 발명에 따른 컴포넌트 식별 장치는, 사용자로부터 객체간의 정보, 이용 케이스의 객체 사용 정보, 알고리즘의 구성 임계값(Component Threshold)(CT), 특정 임계값(Importance Degree Threshold)(IDT)을 입력받는 사용자 인터페이스(GUI)(410), 상기 사용자 인터페이스(410)로부터 정보를 입력받아 객체간의 의존 가중치를 계산하는 가중치 계산기(Weight Calculator)(420), 상기 사용자 인터페이스(410)로부터 입력받은 정보와 상기 가중치 계산기(420)에서 구해진 객체간의 의존 가중치를 이용하여 객체 의존 네트워크를 생성하는 객체 네트워크 생성기(Object

Network Generator)(430), 상기 객체 네트워크 생성기(430)에서 생성된 객체 의존 네트워크를 이용하여 컴포넌트 식별 과정을 제어하는 컴포넌트 식별기(440)(Component Identifier)를 포함한다.

<40> 즉, 본 발명은, 사용자로부터 객체간의 정보, 이용 케이스의 객체 사용 정보, 알고리즘의 구성 임계 값, 특정 임계 값을 입력받아서 객체간의 의존 가중치를 계산하고, 객체 의존성 네트워크를 생성하여 컴포넌트를 식별하는 과정으로 구성되어 있다.

<41> 도 5 는 본 발명에 따른 소프트웨어 컴포넌트 식별 방법에 대한 일실시예 흐름도이다.

<42> 먼저, 객체 모델 정보를 입력받는다(510). 여기서, 상기 객체 모델 정보는 연합(Association), 분해(Decomposition), 일반화(Generalization), 의존(Dependency)으로 크게 분류되며, 이중 일반화(Generalization)와 분해(Decomposition) 정보는 굳건한(tight) 관계를 가지는데 반해 연합(Association)과 의존(Dependency) 관계의 중요도는 경우에 따라 다르므로 일반화(Generalization)와 분해(Decomposition)만을 객체 의존 정보로 기록한다. 연합(Association)과 분해(Dependency)의 가중치는 객체 모델에서는 불분명하므로 이용 케이스에서의 객체 사용 정보를 참조하여 결정한다.

<43> 다음으로, 각 이용 케이스에서 객체 이용 정보를 추출하여 도 2 에 나타난 이용 케이스 및 객체 의존성 그래프를 생성한다(520). 이어서 이용 케이스에서 추출된 객체 이용 정보에 대해 앞서 설명한 가중치 계산 방식을 이용하여 의존 지수를 계산한다(530).

<44> 다음으로, 객체에 연결된 입력 아크(incoming arc)의 가중치를 더하여 객체의 중요 지수(ID)를 계산하고, 상기 중요 지수(ID) 값이 사용자가 미리 정의한 특정 임계 값

(Importance Degree Threshold)을 넘는 객체를 씨드(Seed) 객체로 결정한다(540).

<45> 다음으로, 컴포넌트 식별 과정의 준비 단계로 각 컴포넌트에 씨드(Seed) 객체를 하나씩 설정하고, 추가적으로 객체 네비게이션(navigation)을 수행할 것인지 여부를 판단하기 위하여 각 컴포넌트마다 Done[i]라는 플래그(flag) 조건을 부여한다(550). 여기서, 초기 값으로 Done[i]는 모두 거짓(false)이다. 즉, 모든 컴포넌트에서 초기 객체 네비게이션이 가능하다.

<46> 다음으로, 식별 알고리즘의 종료를 판단하는 부분으로서 임의의 컴포넌트에서 추가로 네비게이션할 수 있는 객체들이 존재하는지 여부를 판단한다(560). 여기서, 네비게이션이 허용되는 객체는 컴포넌트에 포함되어 있지 않는 객체들이다.

<47> 따라서, 상기 판단 결과, 만약 모든 컴포넌트가 네비게이션이 허용되어 있지 않으면 식별 알고리즘은 종료하게 된다.

<48> 상기 판단한 결과, 네비게이션이 가능한 컴포넌트가 존재하면 그 컴포넌트에 연결된, 즉 아직 다른 컴포넌트에 포함되지 않는 객체 중에 컴포넌트와의 의존성 값이 특정 임계 값(Component Threshold) 이상이 되는 객체가 존재하는지 여부를 판단한다(570). 여기서, 상기 특정 임계 값(Component Threshold)은 사용자가 결정하게 되며 임계 값(CT)에 비교되는 상기 의존성 값은 도 3의 아크의 가중치 값을 그대로 쓰지 않고 가중치 값을 소스 객체의 중요 지수(ID)로 나눈 값(Dependency Degree(source, target)/Importance Degree(source))을 이용한다.

<49> 상기 다른 컴포넌트에 포함되지 않는 객체 중에 컴포넌트와의 의존성 값이 특정 임계 값(Component Threshold) 이상이 되는 객체가 존재하는지 여부를 판단한 결과, 존재

하면 해당 객체를 컴포넌트에 추가하고, 추가로 네비게이션할 수 있는 객체가 존재하는지 판단하는 과정(560)으로 진행한다.

<50> 상기 다른 컴포넌트에 포함되지 않는 객체 중에 컴포넌트와의 의존성 값이 특정 임계 값(Component Threshold) 이상이 되는 객체가 존재하는지 여부를 판단한 결과, 존재하지 않으면 연결 가능한 새로운 객체가 존재하지 않는 것이므로 Done[i]를 참(true)으로 설정하여 i번째 컴포넌트에서는 더 이상 네비게이션이 발생하지 않게 하고(580), 추가로 네비게이션할 수 있는 객체가 존재하는지 판단하는 과정(560)으로 진행한다.

<51> 상술한 바와 같은 본 발명의 방법은 프로그램으로 구현되어 컴퓨터로 읽을 수 있는 형태로 기록매체(씨디롬, 램, 롬, 플로피 디스크, 하드 디스크, 광자기 디스크 등)에 저장될 수 있다.

<52> 이상에서 설명한 본 발명은 전술한 실시예 및 첨부된 도면에 의해 한정되는 것이 아니고, 본 발명의 기술적 사상을 벗어나지 않는 범위 내에서 여러 가지 치환, 변형 및 변경이 가능하다는 것이 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에게 있어 명백할 것이다.

【발명의 효과】

<53> 상기한 바와 같은 본 발명은, 영역 모델링(domain modeling) 정보인 객체 모델과 이용 케이스의 순차도로부터 객체 관계 정보 및 객체 이용 정보를 추출하여 체계적인 소프트웨어 컴포넌트 식별 과정을 제공하므로 사용자의 경험과 직관에 의존해 온 식별 과

정을 체계적이고 자동화된 도구로 지원 가능하게 하는 효과가 있다.

<54> 특히, 본 발명은, 종래 방식보다 컴포넌트를 정확히 식별할 수 있는 효과가 있다.

【특허청구범위】**【청구항 1】**

객체 관계와 객체 이용 정보를 이용한 소프트웨어 컴포넌트 식별 장치에 있어서,
사용자로부터 정보를 입력받기 위한 사용자 인터페이싱 수단;

상기 사용자 인터페이싱 수단으로부터 정보를 입력받아 객체간의 의존 가중치를 계산하기 위한 가중치 계산수단;

상기 사용자 인터페이싱 수단으로부터 입력받은 정보와 상기 가중치 계산수단에서 구해진 객체간의 의존 가중치를 이용하여 객체 의존도 네트워크를 생성하기 위한 객체 네트워크 생성수단; 및

상기 객체 네트워크 생성수단에서 생성된 객체 의존도 네트워크를 이용하여 컴포넌트 식별 과정을 제어하기 위한 컴포넌트 식별수단

을 포함하는 소프트웨어 컴포넌트 식별 장치.

【청구항 2】

제 1 항에 있어서,

상기 가중치 계산수단은,

객체간의 의존성을 객체 모델에 나타난 구조적인 관계 뿐만 아니라 유스 케이스(이용 케이스)에 나타나는 객체들간의 누적 이용 정보를 반영하여 하나의 가중치 값으로 나타내고, 이러한 객체 의존성을 이용 케이스의 가중치를 고려하여 계산하는 것을 특징으

로 하는 소프트웨어 컴포넌트 식별 장치.

【청구항 3】

제 1 항에 있어서,

상기 객체 네트워크 생성수단은,

영역 모델링 정보인 객체 모델과 이용 케이스별 순차도에 나타난 정보를 바탕으로 객체들간의 구조적인 연관관계 및 객체 이용 관계를 기술하고 이를 이용하여 양의 실수 값으로 표현되는 객체 의존 지수(Dependency Degree)와 객체 중요 지수(Importance Degree)를 나타내는 것을 특징으로 하는 소프트웨어 컴포넌트 식별 장치.

【청구항 4】

제 1 항에 있어서,

상기 컴포넌트 식별수단은,

상기 객체 네트워크 생성수단에서 생성된 객체 의존도 네트워크에 나타난 객체간의 의존 정도, 객체의 중요 지수, 의존 유형 등을 바탕으로 컴포넌트의 독립성, 재사용성 등을 고려하여 컴포넌트로 나뉘어질 수 있는 객체들로 클러스터링(clustering)하는 것을 특징으로 하는 소프트웨어 컴포넌트 식별 장치.

【청구항 5】

소프트웨어 컴포넌트 식별 장치에 적용되는 컴포넌트 식별 방법에 있어서,

객체 모델 정보를 입력받아 유스 케이스(이용 케이스) 및 객체 의존성 그래프를 생성하는 제 1 단계;

상기 유스 케이스(이용 케이스)에서 추출된 객체 이용 정보에 대하여 객체 의존 지수를 계산하고, 객체에 연결된 입력 아크(incoming arc)의 가중치를 더하여 객체 중요 지수를 계산하는 제 2 단계;

상기 객체 중요 지수 값이 사용자가 미리 정의한 특정 임계 값(Importance Degree Threshold)을 넘는 객체를 씨드(Seed) 객체로 결정하여 컴포넌트에 설정하는 제 3 단계; 및

컴포넌트와의 의존성 값이 특정 임계값 이상이 되는 객체를 컴포넌트에 추가하는 과정을, 추가로 네비게이션이 가능한 객체에 대하여 수행하는 제 4 단계를 포함하는 컴포넌트 식별 방법.

【청구항 6】

제 5 항에 있어서,

상기 제 4 단계는,

추가적인 객체 네비게이션을 위한 초기 조건을 부여하는 제 5 단계;

추가로 네비게이션을 할 수 있는 객체가 존재하는지 확인하는 제 6 단계;

상기 제 6 단계의 확인 결과, 상기 추가로 네비게이션할 수 있는 객체가 존재하지 않으면 종료하고, 상기 추가로 네비게이션할 수 있는 객체가 존재하면 컴포넌트와의 의존성 값이 특정 임계값 이상이 되는 객체가 존재하는지 확인하는 제 7 단계; 및

상기 제 7 단계의 확인 결과, 상기 컴포넌트와의 의존성 값이 특정 임계값 이상이 되는 객체가 존재하면 해당 객체를 컴포넌트에 추가한 후 제 6 단계로 진행하고, 컴포넌트와의 의존성 값이 특정 임계값 이상이 되는 객체가 존재하지 않으면 Done[i]를 true로 설정하고 제 6 단계로 진행하는 제 8 단계

를 포함하는 컴포넌트 식별 방법.

【청구항 7】

제 5 항에 있어서,

상기 제 2 단계는,

영역 모델링 정보인 객체 모델과 이용 케이스별 순차도에 나타난 정보를 바탕으로 객체들간의 구조적인 연관관계 및 객체 이용 관계를 기술하고 이를 이용하여 양의 실수 값으로 표현되는 객체 의존 지수(Dependency Degree)와 객체 중요 지수(Importance Degree)를 계산하는 것을 특징으로 하는 컴포넌트 식별 방법.

【청구항 8】

제 5 항에 있어서,

상기 제 4 단계는,

객체간의 의존 정도, 객체의 중요 지수, 의존 유형 등을 바탕으로 컴포넌트의 독립성, 재사용성 등을 고려하여 컴포넌트로 나뉘어질 수 있는 객체들로 클러스터링

(clustering)하는 것을 특징으로 하는 컴포넌트 식별 방법.

【청구항 9】

대용량 프로세서를 구비한 소프트웨어 컴포넌트 식별 장치에,

객체 모델 정보를 입력받아 유스 케이스(이용 케이스) 및 객체 의존성 그래프를 생성하는 제 1 기능;

상기 유스 케이스(이용 케이스)에서 추출된 객체 이용 정보에 대하여 객체 의존 지수를 계산하고, 객체에 연결된 입력 아크(incoming arc)의 가중치를 더하여 객체 중요 지수를 계산하는 제 2 기능;

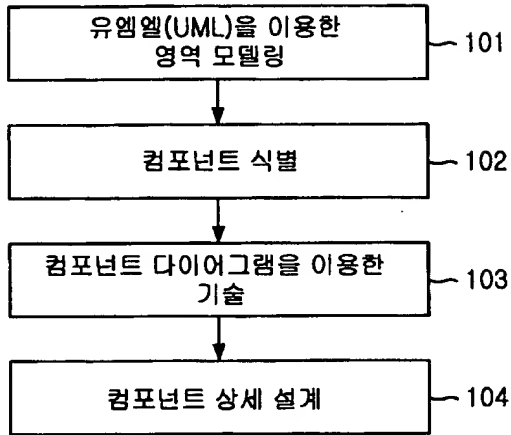
상기 객체 중요 지수 값이 사용자가 미리 정의한 특정 임계 값(Importance Degree Threshold)을 넘는 객체를 씨드(Seed) 객체로 결정하여 컴포넌트에 설정하는 제 3 기능; 및

컴포넌트와의 의존성 값이 특정 임계값 이상이 되는 객체를 컴포넌트에 추가하는 과정을, 추가로 네비게이션이 가능한 객체에 대하여 수행하는 제 4 기능

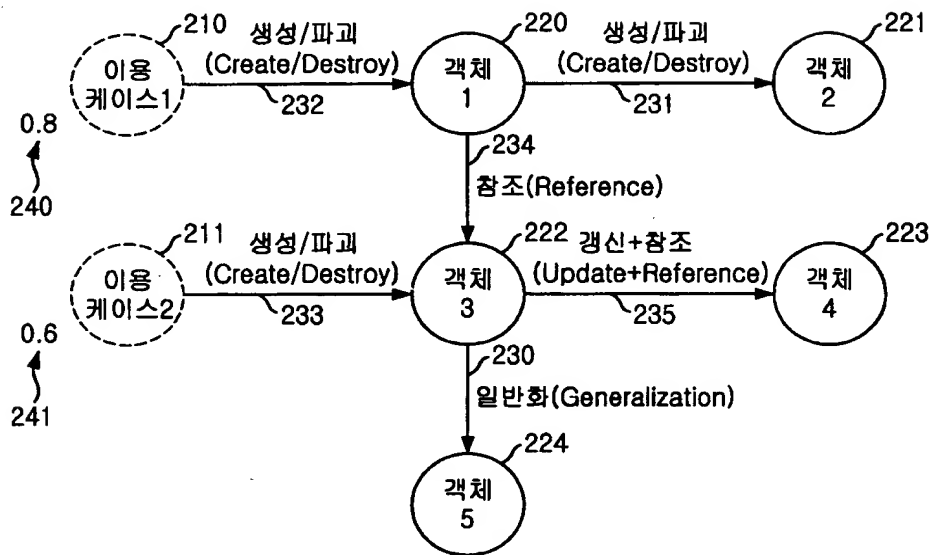
을 실현시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체.

【도면】

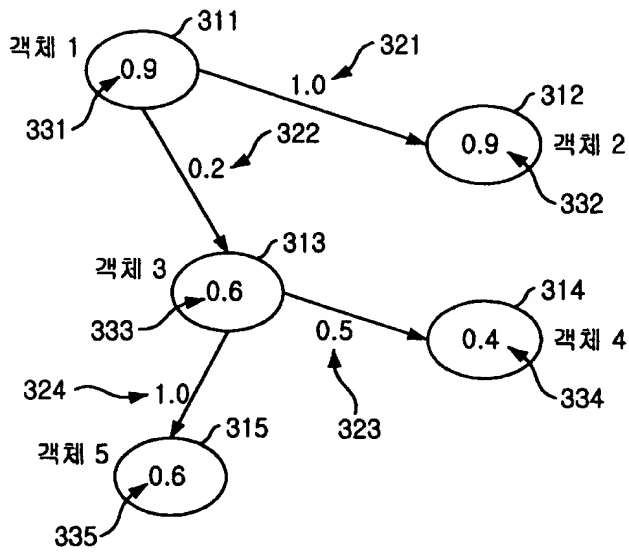
【도 1】



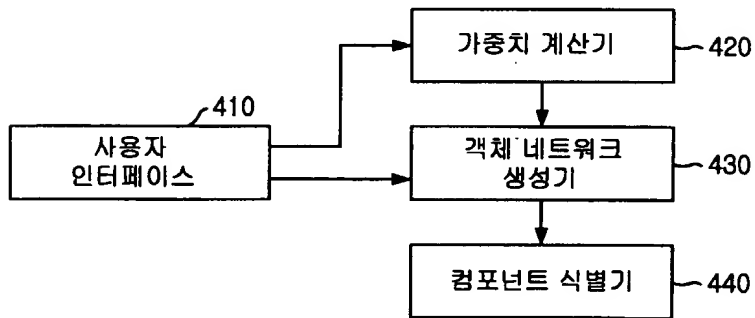
【도 2】



【도 3】



【도 4】



【도 5】

